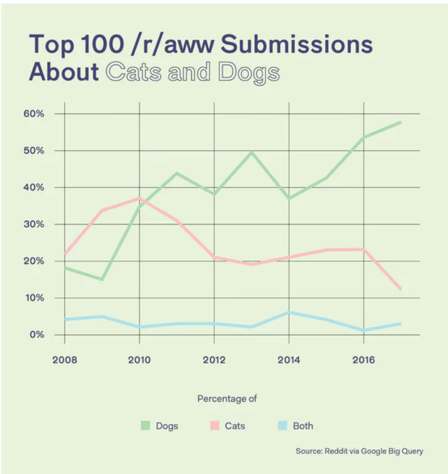
CSE 446: Machine Learning

Sewoong Oh

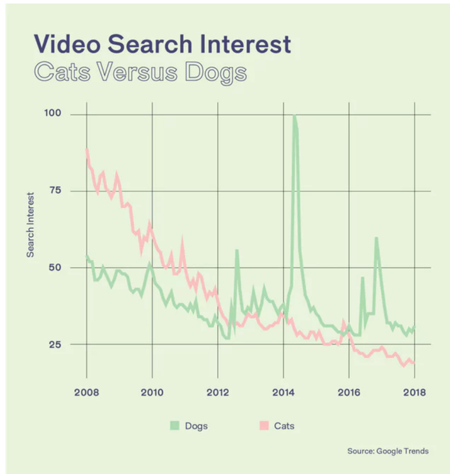
Traditional algorithms vs. Machine Learning

Social media mentions of Cats vs. Dogs

Reddit



Google



Twitter?

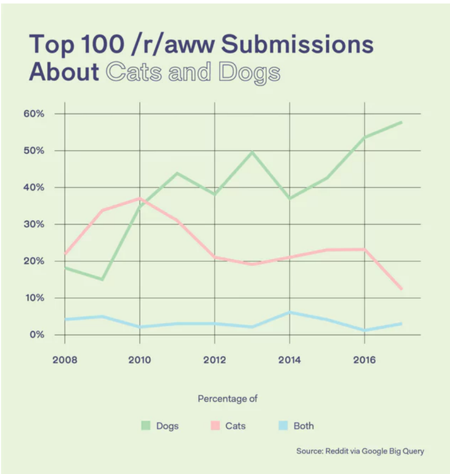
Write a program that sorts tweets into those containing "**cat**", "**dog**", or ***other***

Graphics courtesy of https://theoutline.com/post/3128/dogs-cats-internet-popularity?zd=1

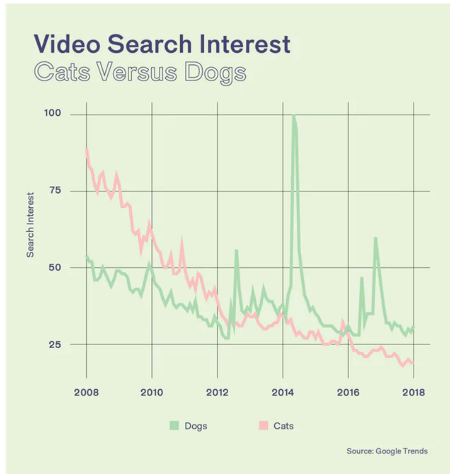
Traditional algorithms

Social media mentions of Cats vs. Dogs

Reddit



Google



Twitter?

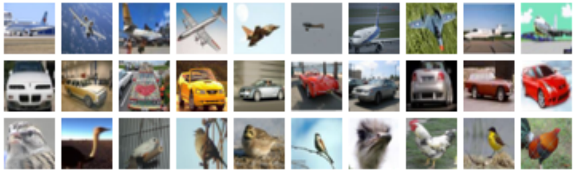
```
cats = []
dogs = []
other = []
for tweet in tweets:
    if "cat" in tweet:
        cats.append(tweet)
    elseif "dog" in tweet:
        dogs.append(tweet)
    else:
        other.append(tweet)
return cats, dogs, other
```

Write a program that sorts tweets into those containing "**cat**", "**dog**", or ***other***

Graphics courtesy of https://theoutline.com/post/3128/dogs-cats-internet-popularity?zd=1

Machine learning algorithms

Write a program that sorts images into those containing "**birds**", "**airplanes**", or ***other.***

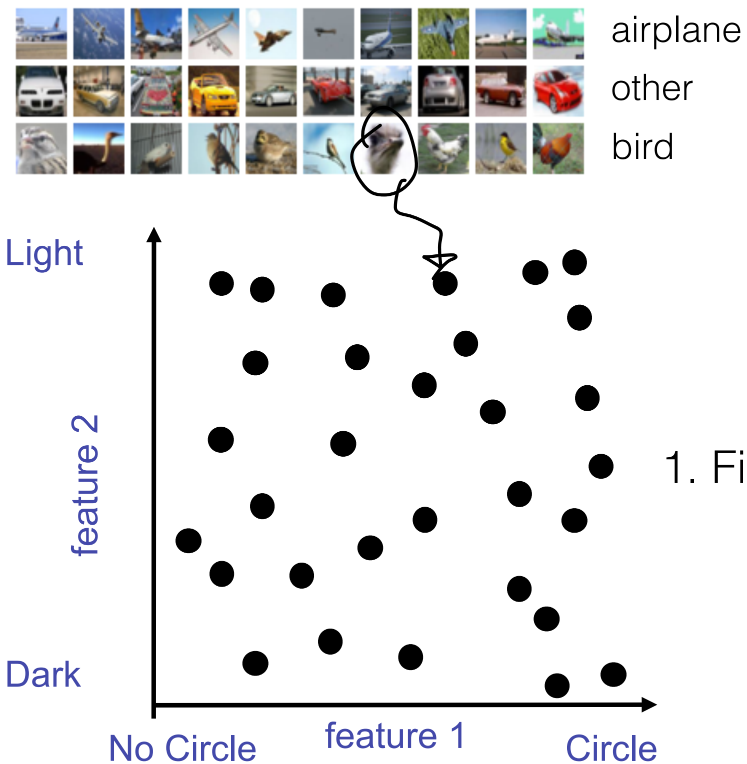



```
birds = []
planes = []
other = []
for image in images:
    if bird in image:
        birds.append(image)
    elseif plane in image:
        planes.append(image)
    else:
        other.append(tweet)
return birds, planes, other
```

Machine learning algorithms

Write a program that sorts images into those containing "**birds**", "**airplanes**", or ***other.***

```
birds = []
planes = []
other = []
for image in images:
    if bird in image:
        birds.append(image)
    elseif plane in image:
        planes.append(image)
    else:
        other.append(tweet)
return birds, planes, other
```

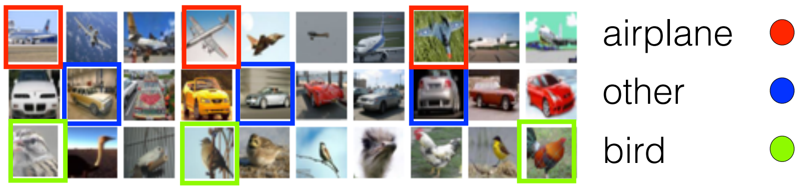


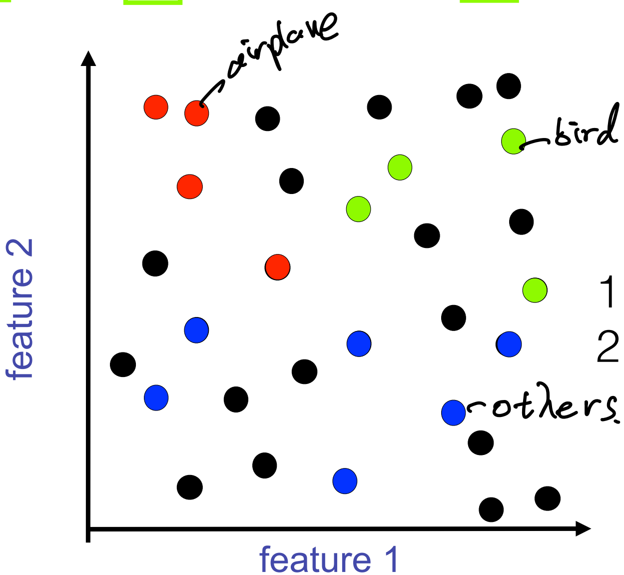
1. Find appropriate representation of the data

Machine learning algorithms

Write a program that sorts images into those containing "**birds**", "**airplanes**", or ***other.***

```
birds = []
planes = []
other = []
for image in images:
    if bird in image:
        birds.append(image)
    elseif plane in image:
        planes.append(image)
    else:
        other.append(tweet)
return birds, planes, other
```



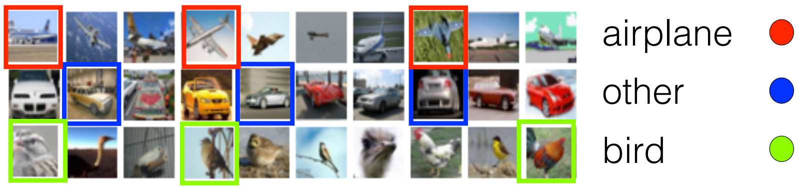


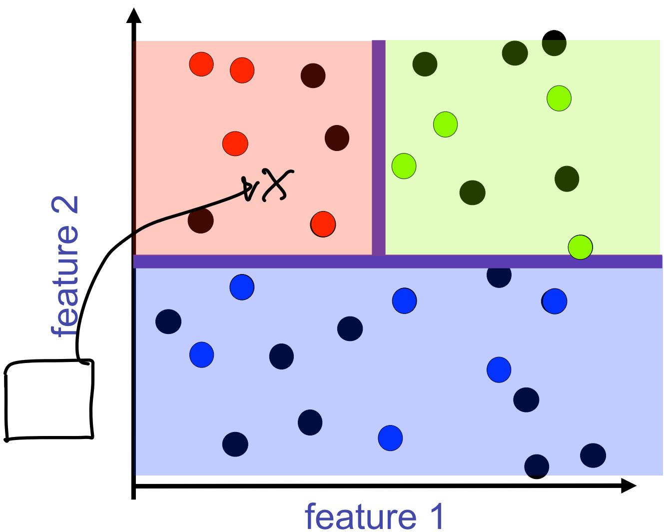
1. Find appropriate representation of the data
2. Crowdsource some samples to get labels

Machine learning algorithms

Write a program that sorts images into those containing "**birds**", "**airplanes**", or ***other.***

```
birds = []
planes = []
other = []
for image in images:
    if bird in image:
        birds.append(image)
    elseif plane in image:
        planes.append(image)
    else:
        other.append(tweet)
return birds, planes, other
```

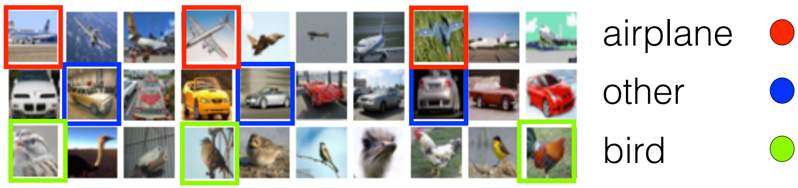


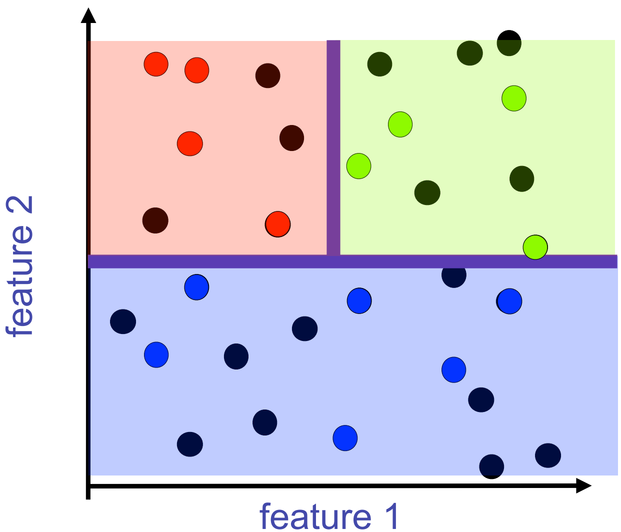


1. Find appropriate representation of the data
2. Crowdsource some samples to get labels
3. Run a machine learning algorithm
   to find decision boundaries

Machine learning algorithms

Write a program that sorts images into those containing "**birds**", "**airplanes**", or ***other.***






```
birds = []
planes = []
other = []
for image in images:
    if bird in image:
        birds.append(image)
    elseif plane in image:
        planes.append(image)
    else:
        other.append(tweet)
return birds, planes, other
```

Traditional Algorithm

The decision rule of

`if "cat" in tweet:`

is **hard coded by expert.**

Machine Learning

The decision rule of

`if bird in image:`

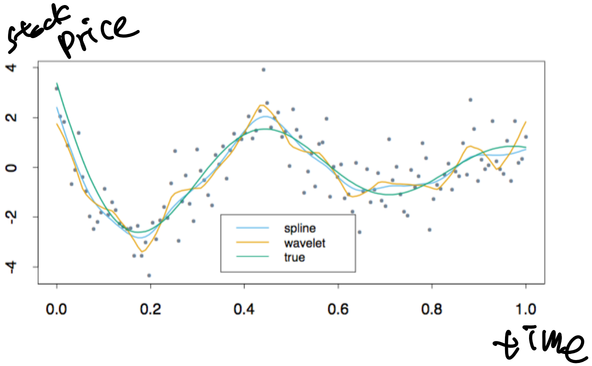
is **LEARNED using DATA**

Machine learning is incredibly powerful and can have significant (unintended) negative consequences on society through targeting, excluding, and misusing.

Learning objectives of this course:

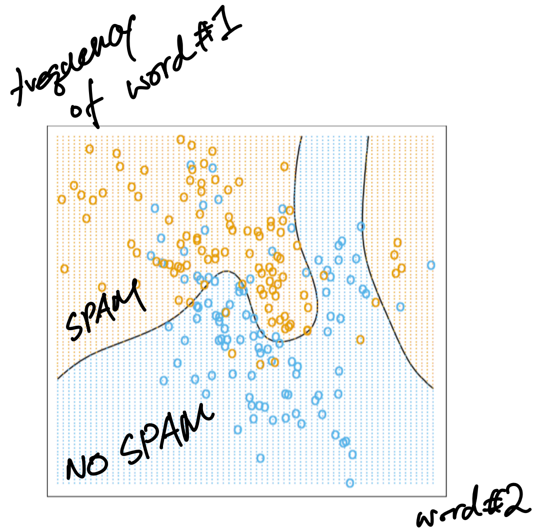
- introduction to the fundamental concepts of machine learning
- analysis and implementation of machine learning algorithms
- knowing how to use machine learning responsibly and robustly

Flavors of ML



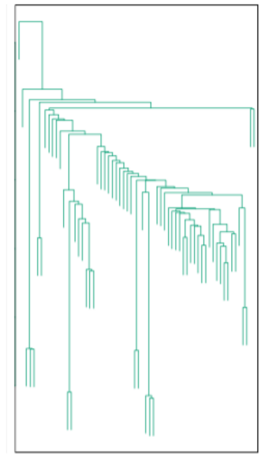
Regression

Predict continuous value:
ex: stock market, credit score, temperature, Netflix rating



Classification

Predict categorical value:
loan or not? spam or not? what disease is this?



Unsupervised Learning

Predict structure:
tree of life from DNA, find similar images, community detection

labelled data = Supervised Learning

unlabelled data

Mix of statistics (theory) and algorithms (programming)

CSE446: Machine Learning

What this class is:

- **Fundamentals of ML:** bias/variance tradeoff, overfitting, optimization and computational tradeoffs, supervised learning (e.g., linear, boosting, deep learning), unsupervised models (e.g. k-means, EM, PCA)
- **Preparation for further learning:** the field is fast-moving, you will be able to apply the basics and teach yourself the latest

What this class is not:

- **Survey course:** laundry list of algorithms, how to win Kaggle
- **An easy course:** familiarity with intro linear algebra and probability are assumed, homework will be time-consuming

Course Logistics

- All the information can be found at Course Website:
  https://courses.cs.washington.edu/courses/cse446/22wi/

- **All zoom links are on Canvas**
  - First week lectures 1-3
  - First week sections
  - OHs

- **Instructor**: Sewoong Oh
- **9 amazing TAs**: Jakub Filipek, Joshua Gardner, Thai Quoc Hoang, Chase King, Tim Li, Pemi Nguyen, **Hugh Sun**, Yuhao Wan, Kyle Zhang
- **Lectures**: MWF 9:30-10:20 (first week on Zoom)
- **Questions/announcements/discussions**: EdStem, link on website
- **Personal questions**: cse446-staff@cs.washington.edu
- **Anonymous feedback**: link on website
- **Office hours**: starts on Tuesday, schedule on the website

Prerequisites

- Formally:
  - Linear algebra in MATH 308
  - Algorithm complexity in CSE 312
  - Probability in STAT 390 or equivalent

- Familiarity with:
  - Linear algebra
    - linear dependence, rank, linear equations, SVD
  - Multivariate calculus
    - Differentiate a multi-variate function
  - Probability and statistics
    - Distributions, marginalization, moments, conditional expectation
  - Algorithms
    - Basic data structures, complexity

- "Can I learn these topics concurrently?"
  - Use HW0 to judge skills
  - See website for review materials!

Grading

- 5 homework (100%=12%+22%+22%+22%+22%)
  - Collaboration is okay but must write who you collaborated with.
  - You can spend an arbitrary amount of time discussing and working out a solution with your listed collaborators, but **do not take notes, photos, or other artifacts of your collaboration**. Erase the board you were working on, and once you're alone, write up your answers yourself.
- NO exams
- Extra credit for submitting the proof of course evaluation in the end
- We will assign random subgroups as PODs to collaborate/discuss (when dust clears)

Homework

- HW 0 is out (**Due next Tuesday Jan 11th Midnight**)
  - Short *review*
  - Work individually, treat as barometer for readiness
- HW 1,2,3,4
  - They are not easy or short. Start early.
- Submit to Gradescope (instructions on the website)
- Regrade requests on Gradescope
  - within 7 days of release of the grade
- **There is no credit for late work, you get 5 late days**
  - if HW1 is late by 23 hours, then you used 1 late day
  - If HW1 is late by 25 hours, then you used 2 late days

Homework

- HW 0 is out (**Due next Tuesday Jan 11th Midnight**)
  - Short *review*
  - Work individually, treat as barometer for readiness
- HW 1,2,3,4
  - They are not easy or short. Start early.
- Submit to Gradescope (instructions on the website)
- Regrade requests on Gradescope
  - within 7 days of release of the grade
- **There is no credit for late work, you get 5 late days**
  - if HW1 is late by 23 hours, then you used 1 late day
  - If HW1 is late by 25 hours, then you used 2 late days

1. All code must be written in Python
2. All written work must be typeset (e.g., LaTeX)

See course website for tutorials and references.

Weekly Sections

- Everyone is enrolled in a 50 minutes in-person section on Thursday.
  - Except for week 1 (on Zoom)
- Taught by very talented TAs.
- You are not required to attend.
- There is no attendance or quiz.
- It is meant to help you understand the lectures better and deeper.

Weekly Sections

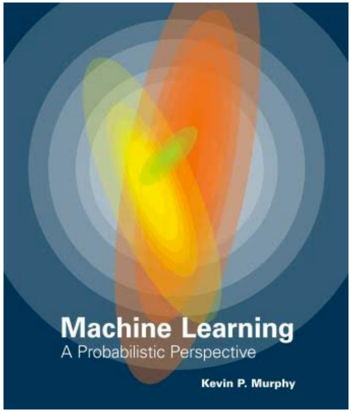
- Previously, We have seen steep decline in attendance in morning sections.
- This time, we have decided to cancel the two morning sections, and instead offer more office hours and dedicate more resources to responding on EdStem

    - Section AA (8:30-9:20): cancelled
    - Section AB (9:30-10:20): cancelled
    - Section AC (10:30-11:20): Chase King, LOW 105
    - Section AD (11:30-12:20): Kyle Zhang, LOW 105
    - Section AE (12:30-1:20): Yuhao Wan, CDH 110B
    - Section AF (1:30-2:20): Jakub Filipek, FSH 107 0

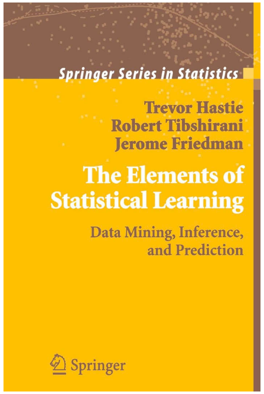
- We ask those registered in AA and AB to attend other sections
- If this is an issue, please contact sewoong@cs.washington.edu

Textbooks

- Required Textbook (optional):
  - ***Machine Learning: a Probabilistic Perspective*; Kevin Murphy**



- Optional Books (free PDF):
  - *The Elements of Statistical Learning: Data Mining, Inference, and Prediction;* Trevor Hastie, Robert Tibshirani, Jerome Friedman



Enjoy!

- ML is becoming ubiquitous in science, engineering and beyond
- It's one of the hottest topics in industry today
- This class should give you the basic foundation for applying ML and developing new methods
- The fun begins…

* Section Locations

Maximum Likelihood Estimation

– How Math helps solve REAL problems.

Your first consulting job

- *Client*: I have a special coin, if I flip it, what's the probability it will be heads?
- *You*: I need to collect ***data***.

H H T T H | ?

Data | Prediction

- *You*: The probability is: $\frac{3}{5}$
- *Client:* Why? What is the principle behind your prediction?

Modelling Coin Flips: Binomial Distribution

- **Data**: sequence $\mathscr{D} = (H, H, T, H, T, \dots)$
  - **k heads** out of **n flips**
- **Hypothesis:** class of models that explains the data.
  - Flips are i.i.d. (independent and identically distributed):
    - Independent events $\mathbb{P}(A \text{ and } B) = \mathbb{P}(A) \cdot \mathbb{P}(B)$
    - Identically distributed according to Bernoulli distribution
      - P(Heads) = θ, P(Tails) = $1 - \theta$
        for some unknown ***parameter*** $\theta \in [0,1]$
- **Generative model:**
  - Probability that the data $\mathscr{D}$ is generated by hypothesis θ is

$$P(\mathscr{D};\theta) = P(HHTHT;\theta)$$

Independance $\rightarrow$ $= \underbrace{P(H;\theta)}_{\theta} \cdot \underbrace{P(H;\theta)}_{\theta} \cdot \underbrace{P(T;\theta)}_{1-\theta} \cdot \underbrace{P(H;\theta)}_{\theta} \cdot \underbrace{P(T;\theta)}_{1-\theta}$

Bernoulli(θ) $\rightarrow$

$$= \theta^k \cdot (1-\theta)^{n-k}$$

Maximum Likelihood Estimation

- **Data**: sequence $\mathscr{D} = (H, H, T, H, T, \ldots)$,
  - **k heads** out of **n flips**
- **Hypothesis:** P(Heads) = θ, P(Tails) = $1 - \theta$
- **Likelihood**:

$$\underbrace{P(\mathscr{D};\theta)}_{\textbf{likelihood}} = \theta^k (1-\theta)^{n-k}$$

- **Maximum likelihood estimation** (MLE): Choose θ that maximizes the probability of observed data:

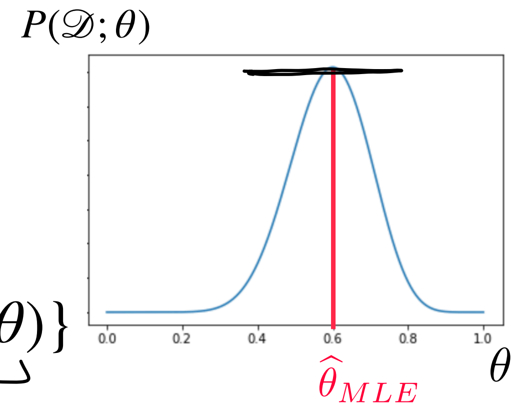
$$\underbrace{\hat{\theta}_{\text{MLE}}}_{\text{Maximum Likelihood Estimate}} = \arg\max_{\theta} P(\mathscr{D};\theta)$$

$$= \arg\max_{\theta} \underbrace{\log P(\mathscr{D};\theta)}_{\text{Log-likelihood}} = \arg\max_{\theta} \; k \cdot \log\theta + (n-k)\log(1-\theta)$$

principled

Your first learning algorithm

$$\widehat{\theta}_{\text{MLE}} = \arg\max_{\theta} \log P(\mathscr{D};\theta)$$
$$= \arg\max_{\theta} \log\{\theta^{k}(1-\theta)^{n-k}\}$$
$$= \arg\max_{\theta} \Big\{\underbrace{k\log\theta + (n-k)\log(1-\theta)}_{\ell(\theta)}\Big\}$$



- Use the fact that derivative is zero at maxima (and also minima)
- Set derivative to zero,

  and find θ satisfying: $$\frac{d}{d\theta}\log P(\mathscr{D};\theta) = 0$$

$$\frac{d\,\ell(\theta)}{d\theta} = \frac{K}{\theta} - \frac{n-K}{1-\theta} = \frac{K - \cancel{K\theta} - n\theta + \cancel{K\theta}}{\theta(1-\theta)}$$

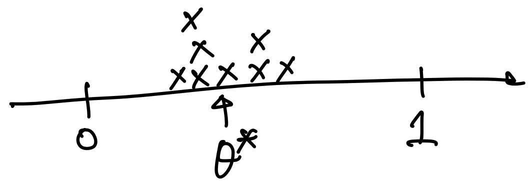
$$= \frac{K - n\theta}{\theta(1-\theta)} = 0$$

find θ such that

$$\boxed{\widehat{\theta}_{MLE} = \frac{K}{n}}$$

How good is MLE?

- We treat MLE $\widehat{\theta}_{\text{MLE}}$ as a random variable, where there is a ground truth parameter θ^* that generates the data $\mathscr{D} = (HHTTH\ldots)$ of a fixed size n



$\hat{\theta}_{MLE} = \frac{k}{n} \leftarrow$ Random

- What can we say about this random variable $\widehat{\theta}_{\text{MLE}}$?
- First good property of MLE for Binomial: **unbiased**
  - Definition: **bias** of our MLE is

$$\text{Bias}(\widehat{\theta}_{\text{MLE}}) := \mathbb{E}_{\mathscr{D}\sim P_{\theta^*}}\left[\hat{\theta}_{\text{MLE}}\right] - \theta^* = \mathbb{E}\left[\frac{k}{n}\right] - \theta^*$$

$$= \theta^* - \theta^* = 0$$

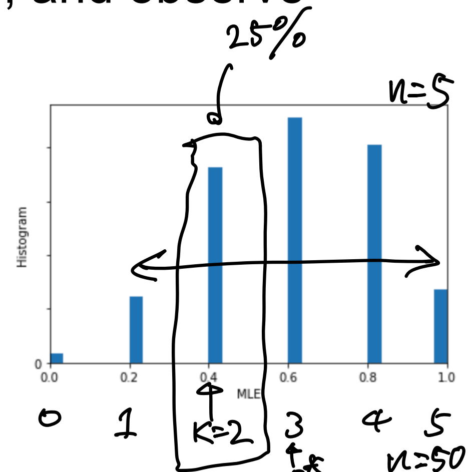
1st order statistic

- **Expectation** describes how the estimator behaves *on average*

How many flips do I need?

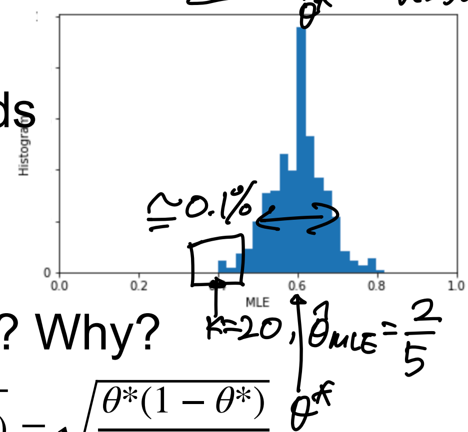
- Consider running many experiments with $\theta^* = \frac{3}{5}$, and observe many instances of the random variable

$$\widehat{\theta}_{MLE} = \frac{k}{n}$$



- *Client*: I flipped the coin 5 times and got 2 heads.

$$\widehat{\theta}_{MLE} = \frac{2}{5}$$



- *Client*: I flipped the coin 50 times and got 30 heads

$$\widehat{\theta}_{MLE} = \frac{30}{50} = \frac{3}{5}$$

- *Client:* they are both unbiased, which one is right? Why?
- The width of typical uncertainty is about $\sqrt{\mathrm{Var}(\widehat{\theta}_{\mathrm{MLE}})} = \sqrt{\frac{\theta^*(1-\theta^*)}{n}}$

Quantifying Uncertainty

2nd order statistics

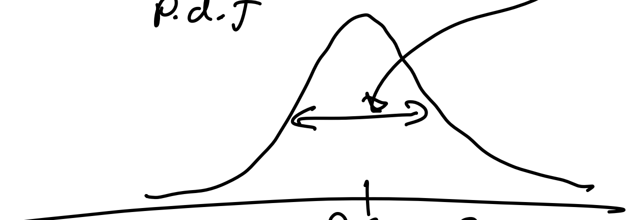
- The **Variance** is the expected squared deviation from the mean:

$$\text{Variance}(\widehat{\theta}_{MLE}) := \mathbb{E}\left[\left(\widehat{\theta}_{MLE} - \mathbb{E}[\widehat{\theta}_{MLE}]\right)^2\right]$$

- As a rule of thumb

$$\widehat{\theta}_{\text{MLE}} \simeq \mathbb{E}[\widehat{\theta}_{\text{MLE}}] \pm \sqrt{\text{Variance}(\widehat{\theta}_{\text{MLE}})}$$

p.d.f



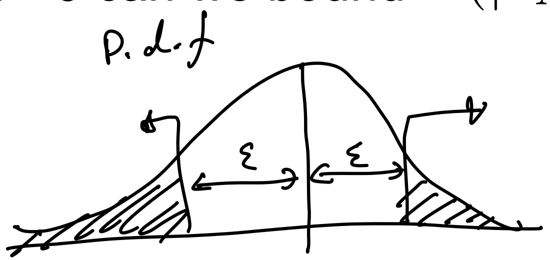
$\mathbb{E}[\hat{\theta}_{MLE}]$

$n \uparrow \infty$

- Second good property of MLE: **minimum (asymptotic) variance**
  i.e, for all estimators $\widehat{\theta}$, $\lim\limits_{n\to\infty} \text{Var}(\widehat{\theta}_{\text{MLE}}) \leq \lim\limits_{n\to\infty} \text{Var}(\widehat{\theta})$

Expectation versus High Probability

- Tail bound of a random variable
- For any $\epsilon > 0$ can we bound $\mathbb{P}(|\widehat{\theta}_{MLE} - \mathbb{E}[\widehat{\theta}_{MLE}]| \geq \epsilon)$?



Markov's inequality

For any $t > 0$ and non-negative random variable X

$$\mathbb{P}(X \geq t) \leq \frac{\mathbb{E}[X]}{t}$$

- **Exercise**: Apply Markov's inequality to obtain bound.
(Hint: set $X = \left| \widehat{\theta}_{\text{MLE}} - \mathbb{E}[\widehat{\theta}_{\text{MLE}}] \right|^2$) → Chebyshev's inequality

Maximum Likelihood Estimation

- **Observe** $X_1, X_2, \ldots, X_n$ drawn i.i.d. from $P(X_i; \theta)$ for some true $\theta = \theta^*$
- **Likelihood function**: $L_n(\theta) = \prod_{i=1}^{n} P(X_i; \theta)$
- **Log-likelihood function**: $\ell_n(\theta) = \log L_n(\theta) = \sum_{i=1}^{n} \log P(X_i; \theta)$
- **Maximum Likelihood Estimator (MLE)**: $\widehat{\theta}_{\mathrm{MLE}} = \arg\max_{\theta} \ell_n(\theta)$

Questions?

Questions?

Questions?